REMARKS/ARGUMENTS

Applicant responds herein to the Office Action dated April 4, 2007.

Claims 1-37 are currently pending in the present application. Claims 1-37 were rejected in the Office Action. Applicant amended Claims 1, 20 and 37 and respectfully request a reconsideration of the rejection.

Rejection under 35 U.S.C. §112.

Claims 1-37 were rejected under 35 U.S.C. §112, second paragraph because of the indefiniteness of the limitation "some node" previously recited in the independent claims. Applicant corrected this informality by taking the Examiner's suggestion and amended Claims 1, 20 and 37 to recite the limitation of "one of said nodes."

Rejection under 35 U.S.C. §§102 and 103.

Claims 1-8, 18, 20-27, 35 and 37 were rejected in the Office Action under 35 U.S.C. §102(b) as being anticipated by the article entitled "Improving the Granularity of Access Control for Windows 2000" (hereinafter "Swift"). Claims 9-17 and 28-34 were rejected under 35 U.S.C. §103(a) as being unpatentable over Swift in view of U.S. Patent Publication No. 2004/0186845 (hereinafter "Fukui"). Finally, Claims 19 and 36 were rejected under 35 U.S.C. §103(a) as being unpatentable over Swift in view of U.S. Patent Publication No. 2003/0187854 (hereinafter "Fairweather"). Applicant respectfully disagrees and requests reconsideration of the rejections.

Independent Claims 1, 20 and 30 recite the method, apparatus and a program of instructions, respectively, for managing availability of information stored in a tree structure "including a plurality of nodes sequentially arranged from a home root node to at least one leaf node." Further, each independent claim recites a condition requiring "that the number of times of changes in the availability condition is limited to one at maximum on any of paths from said home root node to said respective leaf nodes." These limitations of Claims 1, 20 and 37 are not disclosed in the cited prior art.

Swift discloses the mechanisms in Windows 2000 that enable fine-grained and centrally managed access control for operating system components and applications. An example of the hierarchy of information discussed in Swift is shown in Fig. 5. Here, the "Company" container represents the home root node, and the "Jane User" object represents a leaf node. Each object and

each container has a set of properties. Swift teaches that there are properties that are common to many types of objects and that, instead of giving access right to a particular object, users may be given access to particular properties of an object. This is particularly illustrated in the Example on pages 14-15. Specifically, the first ACE grants administrators full control over this user object, the second ACE grants group administrators read and write access to user's public information, and the third ACE grants a user herself an access to change the password. Accordingly, access to a particular set of properties is controlled by this set's administrator. In other words, each node of Fig. 5 may have several administrators depending on the properties of this node.

Moreover, while Swift discloses controlling access to a particular property set from a single place in the directory, it does not disclose or even suggest that along a path from the home root node to the leaf node the availability condition changes not more than once. In fact, it might not be possible in the system taught by Swift. For example, applying the methodology disclosed in Swift to the system of its Fig. 5, the "Company" node has properties that will only be accessible by the company administrators, the "Departments" node has properties that will only be accessible by company administrators and department administrators, the "Research" node has properties that will only be accessible by company administrators, department administrators and group administrators, and, finally, the Jane User node has properties that will only be accessible by company administrators, department administrators, group administrators and Jane User. These multiple changes in access availability is possible because, as Swift repeatedly indicates, "despite presenting data as a hierarchy, the Active Directory internally stores data in a flat database and maintains indexes" over the names and properties of the objects. Therefore, the number of times the availability condition changes along a particular path is irrelevant to the system disclosed in Swift.

In view of the above, Applicant respectfully submits that the limitation of Claims 1, 20 and 37 requiring that the number of times the availability condition changes on any path from a home root node to a particular leaf node is limited to one at maximum, is not disclosed or suggested by Swift. Further, none of the other cited prior art references remedy this deficiency of

Swift. Accordingly Claims 1, 20 and 37 are allowable over the cited prior art. Moreover, Claims 2-19 and 21-36 depend directly or indirectly on independent Claims 1 and 20, respectively. Therefore, Claims 2-19 and 21-36 are allowable for at least the same reasons as Claims 1 and 20, and, further, on their own merits.

In view of the foregoing discussion, allowance of Claims 1-37 is respectfully requested.

Accordingly, the Examiner is respectfully requested to reconsider the application, allow the claims as amended and pass this case to issue.

Respectfully submitted,

THIS CORRESPONDENCE IS BEING SUBMITTED ELECTRONICALLY THROUGH THE UNITED STATES PATENT AND TRADEMARK OFFICE EFS FILING SYSTEM ON JUNE 29, 2007

MAX MOSKOWITZ
Registration No.: 30,576
OSTROLENK, FABER, GERB & SOFFEN, LLP
1180 Avenue of the Americas
New York, New York 10036-8403
Telephone: (212) 382-0700